

**AIM:** To demonstrate various queuing mechanisms and make comparative analysis of various queuing techniques. (using trace file-using bus topology) (Droptail, RED, SFQ, FQ)

**Code:(TCP + UDP)**

```
set ns [new Simulator]
#define color for data flows
$ns color 1 Blue
$ns color 2 Red
#open tracefiles
set tracefile1 [open bus_tcpudp_sfq.tr w]
$ns trace-all $tracefile1
#open nam file
set namfile [open bus_tcpudp_sfq.nam w]
$ns namtrace-all $namfile
#define the finish procedure
proc finish {} {
    global ns tracefile1 namfile
    $ns flush-trace
    close $tracefile1
    close $namfile
    exec nam bus_tcpudp_sfq.nam &
    exit 0
}
#create six nodes
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]
set n4 [$ns node]
set n5 [$ns node]

#create links between the nodes
#$ns duplex-link $n0 $n0 2Mb 10ms DropTail
#$ns duplex-link $n1 $n0 2Mb 10ms DropTail
$ns simplex-link $n0 $n1 0.3Mb 10ms SFQ
$ns simplex-link $n1 $n0 0.3Mb 10ms SFQ
set lan [$ns newLan "$n1 $n2 $n3 $n4 $n5" 0.5Mb 20ms LL Queue/SFQ MAC/Csma/Ca Channel]

#Give node position
#$ns duplex-link-op $n0 $n0 orient right-down
#$ns duplex-link-op $n1 $n0 orient right-up
$ns simplex-link-op $n0 $n1 orient right
$ns simplex-link-op $n1 $n0 orient left

#set queue size of link(n0-n1) to 10
$ns queue-limit $n0 $n1 10

#setup TCP connection
set tcp [new Agent/TCP/Newreno]
$ns attach-agent $n0 $tcp
set sink [new Agent/TCPSink/DelAck]
$ns attach-agent $n2 $sink
```

```
$ns connect $tcp $sink
$tcp set fid_ 1
$tcp set packet_size_ 1000

#set ftp over tcp connection
set ftp [new Application/FTP]
$ftp attach-agent $tcp

#Setup a UDP connection
set udp [new Agent/UDP]
$ns attach-agent $n3 $udp
set null [new Agent/Null]
$ns attach-agent $n4 $null
$ns connect $udp $null
$udp set fid_ 2
$udp set packet_size_ 1000

#Setup a CBR over UDP connection
set cbr [new Application/Traffic/CBR]
$cbr attach-agent $udp
$cbr set type_ CBR
$cbr set packet_size_ 1000
$cbr set rate_ 1mb
$cbr set random_ false

#scheduling the events

$ns at 0.1 "$ftp start"
$ns at 2.5 "$cbr start"
$ns at 4.9 "$ftp stop"
$ns at 5.0 "$cbr stop"

$ns at 5.5 "finish"
$ns run
```

**Output:****TCP+UDP(DropTail)**

```
shinchan@Lenovo-ideapad-300-15ISK:~/Study/D/5th Sem/CN lab/Bus topology$ gawk -f Droppacket.awk bus.
.tr
Packet Sent:2561
Packet Received:835
Packet Delivery Ratio:306.71

Drop Packets:1726
```

**TCP+UDP(RED)**

```
shinchan@Lenovo-ideapad-300-15ISK:~/Study/D/5th Sem/CN lab/Bus topology$ gawk -f Droppacket.awk bus.
.tr
Packet Sent:2468
Packet Received:797
Packet Delivery Ratio:309.66

Drop Packets:1671
```

**TCP+UDP(SFQ)**

```
shinchan@Lenovo-ideapad-300-15ISK:~/Study/D/5th Sem/CN lab/Bus topology$ gawk -f Droppacket.awk bus.  
tr  
Packet Sent:2561  
Packet Received:835  
Packet Delivery Ratio:306.71  
  
Drop Packets:1726
```

**TCP+UDP(FQ)**

```
shinchan@Lenovo-ideapad-300-15ISK:~/Study/D/5th Sem/CN lab/Bus topology$ gawk -f Droppacket.awk bus.  
tr  
Packet Sent:2593  
Packet Received:849  
Packet Delivery Ratio:305.42  
  
Drop Packets:1744
```

**Code:(TCP)**

```
set ns [new Simulator]  
#define color for data flows  
$ns color 1 Blue  
$ns color 2 Red  
  
#open tracefiles  
set tracefile1 [open bus_tcp_droptail.tr w]  
$ns trace-all $tracefile1  
#open nam file  
set namfile [open bus_tcp_droptail.nam w]  
$ns namtrace-all $namfile  
#define the finish procedure  
proc finish { } {  
    global ns tracefile1 namfile  
    $ns flush-trace  
    close $tracefile1  
    close $namfile  
    exec nam bus_tcp_droptail.nam &  
    exit 0  
}  
#create six nodes  
#set n0 [$ns node]  
#set n1 [$ns node]  
set n0 [$ns node]  
set n1 [$ns node]  
set n2 [$ns node]  
set n3 [$ns node]  
set n4 [$ns node]  
set n5 [$ns node]  
  
#create links between the nodes  
#$ns duplex-link $n0 $n0 2Mb 10ms DropTail  
#$ns duplex-link $n1 $n0 2Mb 10ms DropTail  
$ns simplex-link $n0 $n1 0.3Mb 10ms DropTail  
$ns simplex-link $n1 $n0 0.3Mb 10ms DropTail
```

```

set lan [$ns newLan "$n1 $n2 $n3 $n4 $n5" 0.5Mb 20ms LL Queue/DropTail MAC/Csma/Cd
Channel]

#Give node position
#$ns duplex-link-op $n0 $n0 orient right-down
#$ns duplex-link-op $n1 $n0 orient right-up
$ns simplex-link-op $n0 $n1 orient right
$ns simplex-link-op $n1 $n0 orient left

#set queue size of link(n0-n1) to 10
$ns queue-limit $n0 $n1 10

#setup TCP connection
set tcp [new Agent/TCP/Newreno]
$ns attach-agent $n0 $tcp
set sink [new Agent/TCPSink/DelAck]
$ns attach-agent $n2 $sink
$ns connect $tcp $sink
$tcp set fid_ 1
$tcp set packet_size_ 1000

#set ftp over tcp connection
set ftp [new Application/FTP]
$ftp attach-agent $tcp

#scheduling the events

$ns at 0.1 "$ftp start"
$ns at 5.0 "$ftp stop"

$ns at 5.5 "finish"
$ns run

```

**Output:****TCP(DropTail)**

```

shinchang@Lenovo-ideapad-300-15ISK:~/Study/D/5th Sem/CN lab/Bus topology$ gawk -f Droppacket.awk bus_
tcp.tr
Packet Sent:1323
Packet Received:526
Packet Delivery Ratio:251.52

Drop Packets:797

```

**TCP(RED)**

```

shinchang@Lenovo-ideapad-300-15ISK:~/Study/D/5th Sem/CN lab/Bus topology$ gawk -f Droppacket.awk bus_
tcp.tr
Packet Sent:1247
Packet Received:496
Packet Delivery Ratio:251.41

Drop Packets:751

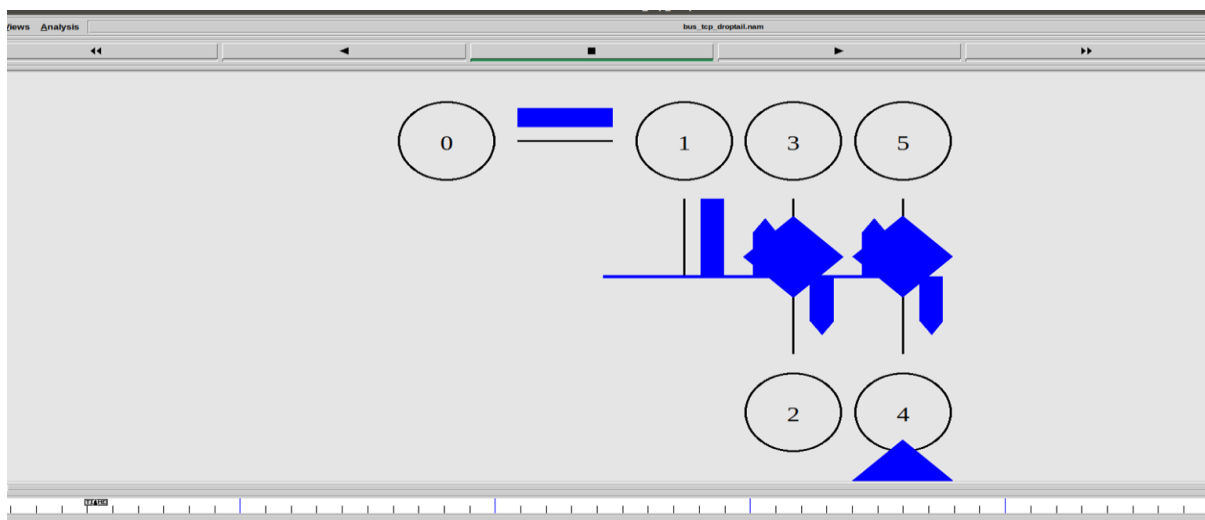
```

## TCP(SFQ)

```
shinchan@Lenovo-ideapad-300-15ISK:~/Study/D/5th Sem/CN lab/Bus topology$ gawk -f Droppacket.awk bus_
tcp.tr
Packet Sent:1339
Packet Received:534
Packet Delivery Ratio:250.75
Drop Packets:805
```

## TCP(FQ)

```
shinchan@Lenovo-ideapad-300-15ISK:~/Study/D/5th Sem/CN lab/Bus topology$ gawk -f Droppacket.awk bus_
tcp.tr
Packet Sent:1366
Packet Received:546
Packet Delivery Ratio:250.18
Drop Packets:820
```



## Code:(udp)

```
set ns [new Simulator]
#define color for data flows
$ns color 1 Blue
$ns color 2 Red
#open tracefiles
set tracefile1 [open bus_udp_red.tr w]
$ns trace-all $tracefile1
#open nam file
set namfile [open bus_udp_red.nam w]
$ns namtrace-all $namfile
#define the finish procedure
proc finish { } {
    global ns tracefile1 namfile
    $ns flush-trace
    close $tracefile1
    close $namfile
    exec nam bus_udp_red.nam &
    exit 0
}
```

```
#create six nodes
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]
set n4 [$ns node]
set n5 [$ns node]
#create links between the nodes
#$ns duplex-link $n0 $n0 2Mb 10ms RED
#$ns duplex-link $n1 $n0 2Mb 10ms RED
$ns simplex-link $n0 $n1 0.3Mb 10ms SFQ
$ns simplex-link $n1 $n0 0.3Mb 10ms SFQ
set lan [$ns newLan "$n1 $n2 $n3 $n4 $n5" 0.5Mb 20ms LL Queue/SFQ MAC/Csma/Ca Channel]

#Give node position
#$ns duplex-link-op $n0 $n0 orient right-down
#$ns duplex-link-op $n1 $n0 orient right-up
$ns simplex-link-op $n0 $n1 orient right
$ns simplex-link-op $n1 $n0 orient left

#set queue size of link(n0-n1) to 20
$ns queue-limit $n0 $n1 10

#Setup a UDP connection
set udp [new Agent/UDP]
$ns attach-agent $n3 $udp
set null [new Agent/Null]
$ns attach-agent $n4 $null
$ns connect $udp $null
$udp set fid_ 2
$udp set packet_size_ 1000

#Setup a CBR over UDP connection
set cbr [new Application/Traffic/CBR]
$cbr attach-agent $udp
$cbr set type_ CBR
$cbr set packet_size_ 1000
$cbr set rate_ 1mb
$cbr set random_ false

#scheduling the events
$ns at 0.1 "$cbr start"
$ns at 5.0 "$cbr stop"

$ns at 5.5 "finish"
$ns run
```

**Output:****UDP(DropTail)**

```
shinchan@Lenovo-ideapad-300-15ISK:~/Study/D/5th Sem/CN lab/Bus topology$ gawk -f Droppacket.awk bus_udp.tr
Packet Sent:2453
Packet Received:613
Packet Delivery Ratio:400.16

Drop Packets:1840
```

**UDP (RED)**

```
shinchan@Lenovo-ideapad-300-15ISK:~/Study/D/5th Sem/CN lab/Bus topology$ gawk -f Droppacket.awk bus_udp.tr
Packet Sent:2453
Packet Received:613
Packet Delivery Ratio:400.16

Drop Packets:1840
```

**UDP (SFQ)**

```
shinchan@Lenovo-ideapad-300-15ISK:~/Study/D/5th Sem/CN lab/Bus topology$ gawk -f Droppacket.awk bus_udp.tr
Packet Sent:2453
Packet Received:613
Packet Delivery Ratio:400.16

Drop Packets:1840
```

**UDP (FQ)**

```
shinchan@Lenovo-ideapad-300-15ISK:~/Study/D/5th Sem/CN lab/Bus topology$ gawk -f Droppacket.awk bus_udp.tr
Packet Sent:2453
Packet Received:613
Packet Delivery Ratio:400.16

Drop Packets:1840
```

**Data Table:****1. TCP+UDP**

Queue Type	Sent Package	Receive Package	Drop Package
DropTail	2560	833	1727
RED	2468	797	1671
FQ	2593	849	1744
SFQ	2561	835	1726

**2. TCP**

Queue Type	Sent Package	Receive Package	Drop Package
DropTail	1323	526	797
RED	1247	496	751
FQ	1366	546	820
SFQ	1349	534	805

**3. UDP**

Queue Type	Sent Package	Receive Package	Drop Package
DropTail	2453	613	1840
RED	2453	613	1840
FQ	2453	613	1840
SFQ	2453	613	1840

	TCP	UDP	TCP+UDP
Queue type	# Drop packet	#Drop packet	#Drop packet
Droptail	797	1840	1727
RED	751	1840	1671
FQ	820	1840	1744
SFQ	805	1840	1726

**Conclusion:**

We successfully designed and learned simple tcl script for wired topology of 6 nodes in NS-2 and analysed various tcl parameters like network nodes, links, queues and topology.

We have observed in UDP that even after changing all the parameters #Drop packets remained same. So we would relate it to something like saturation point i.e. the point after which no changes are reflected.



**Question & Answer:****1) Explain Bus topology?****Answer:**

- Alternatively referred to as a line topology, a bus topology is a network setup in which each computer and network device are connected to a single cable or backbone.
- Depending on the type of network card used in each computer of the bus topology, a coaxial cable or a RJ-45 network cable is used to connect computers together.
- The following sections contain both the advantages and disadvantages of using a bus topology with your devices.

➤ Advantages of bus topology:

- It works well when you have a small network.
- It's the easiest network topology for connecting computers or peripherals in a linear fashion.
- It requires less cable length than a star topology.

➤ Disadvantages of bus topology:

- It can be difficult to identify the problems if the whole network goes down.
- It can be hard to troubleshoot individual device issues.
- Bus topology is not great for large networks.
- Terminators are required for both ends of the main cable.
- Additional devices slow the network down.
- If a main cable is damaged, the network fails or splits into two.